IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

SPECIAL SECTION ON SECURITY AND PRIVACY FOR VEHICULAR NETWORKS

Received June 2, 2018, accepted June 29, 2018, date of publication July 10, 2018, date of current version August 7, 2018.

Digital Object Identifier 10.1109/ACCESS.2018.2854842

Spatio-Temporal Network Traffic Estimation and Anomaly Detection Based on Convolutional Neural Network in Vehicular Ad-Hoc Networks

LAISEN NIE[1], (Member, IEEE), YONGKANG LI[1], AND XIANGJIE KONG[2], (Senior Member, IEEE)

[1]School of Electronics and Information, Northwestern Polytechnical University, Xi'an 710072, China
[2]School of Software, Dalian University of Technology, Dalian 116620, China

Corresponding authors: Laisen Nie (nielaisen@nwpu.edu.cn) and Xiangjie Kong (xjkong@ieee.org)

This work was supported in part by the National Natural Science Foundation of China under Grant 61701406 and Grant 61701413, in part by the Fundamental Research Funds for the Central Universities under Grant 3102017OQD022, and in part by the Natural Science Foundation of Shaanxi Province under Grant 2018JQ6095.

ABSTRACT Over the last decade, vehicular ad-hoc networks (VANETs) have received a greater attention in academia and industry due to their influence in intelligent transportation systems. Providing reliability and security to the VANET is essential in order to guarantee the efficiency of its applications. Anomaly detection has become a challenging problem due to the unique environment of VANETs with quick movement and short-lived link. In this paper, a method using the spatio-temporal feature of network traffic is proposed to implement network traffic estimation at first, and on this basis, an anomaly detection algorithm is put forward. The convolutional neural network is employed to extract the spatio-temporal features of the traffic matrix. In terms of the extracted features, network traffic is estimated by using a fully connected architecture as the output layer. Then, a threshold-based separation method is used to implement anomaly detection. The preliminary experiments comparing the proposed method with other machine learning-based methods show the effectiveness of the proposed anomaly detection method.

INDEX TERMS Vehicular ad-hoc network, anomaly detection, network traffic, convolutional neural network.

I. INTRODUCTION

Vehicular Ad-Hoc Networks (VANET), as a representative paradigm of mobile ad-hoc networks (MANETs), are potential to enhance road safety and provide traffic efficiency for future Intelligent Transportation Systems (ITS). A typical VANET consists of users (the key component of VANETs) and infrastructures of networking for corresponding users. Hence, VANETs include two types of communications, that is, Vehicle-to-Vehicle (V2V) and Vehicle-to-Infrastructure (V2I). Simultaneously, it provides safety-related and non-safety applications (e.g., collision warning and infotainment respectively). To guarantee users' privacy and security, the malware, errors and intrusions are detected by security operators according to the real-time detection of anomalous behaviors [1], [2]. Detecting abnormal behaviors of network traffic is crucial for the network operators to provide the available services for users [3]–[7], because a series of operations are operated to make the network normal again after detecting the volume anomalies. Although many efforts have been made to against attacks from all possible malicious users, there are still many challenges caused by the characteristics of VANET such as quick movement and short-lived link et al.

In traditional ISP networks, time series methods viewed as threshold-based methods are ubiquitous [8]–[12]. These methods first model network traffic by various statistical models for traffic estimation [11]. Furthermore, a threshold is usually employed to distinguish normal and abnormal behaviors of traffic flows. Nevertheless, these per-flow methods have not been competent for anomaly detection in VANETs due to its distinguishing characteristics in networking. A typical architecture of VANETs is shown in Fig. 1. The road side unit (RSU) provides Internet services for users (namely, On Board Unit) by wireless links. The differences between

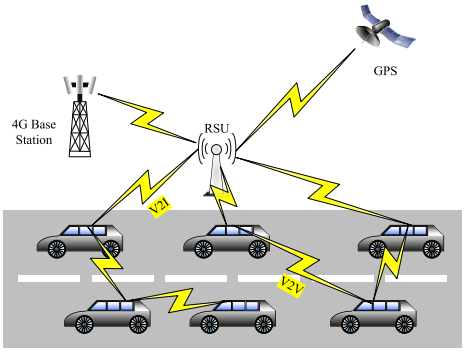


FIGURE 1. VANET architecture.

traditional ISP networks and VANETs in networking are reflected in the user's behaviors including relative location and dynamical access.

- In VANETs, each user moves quickly all the time, which leads to the relative distance between any two users is dynamical. The architecture of traditional ISP networks is statical comparing with VANETs for a while. Under this case, the links are redirected frequently, and the topology of a VANET changes rapidly.
- The users of a VANET access network randomly, which leads to the short-lived link. The short-lived link also causes the dynamical topology of VANETs with small timescale.

Anomaly detection algorithms for traditional ISP networks aim at the network traffic with large timescale. Generally, the timescale of network traffic in traditional ISP network is 10-min or 15-min [13]–[17]. Unfortunately, most of links in VANETs cannot maintain such long time. Considering the statistical characteristics of traffic flows, the network traffic of a VANET exhibits much more irregular fluctuations. Nevertheless, the ISP network traffic shows large-range dependence, multifractal feature and a small number of fluctuations [17]. For instance, Fig. 2 plots the traffic flows of the Abilene backbone network. Obviously, we observe that these traffic flows exhibit regular traces. That is because the Abilene network traffic is sampled by 10-min timescale, and the dataset with 2016 time slots reveals the volume of

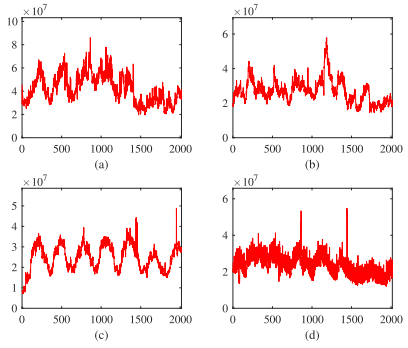


FIGURE 2. Traffic flows from the Abilene backbone network.

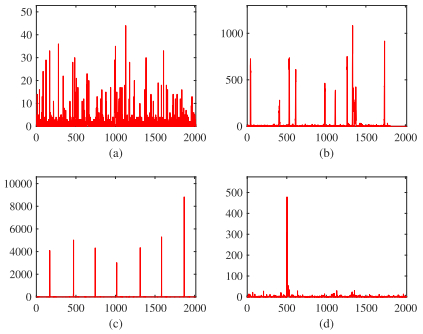


FIGURE 3. Traffic flows from our testbed.

traffic during a week. Namely, these traffic traces depend on the behaviors of users explicitly, and the outlines of traffic express the rest and work period roughly. By contrast, the traffic flows of VANETs with short timescale (1-sec) shown by Fig. 3 have much more irregular fluctuations. Capturing short-range traffic flows, especially irregular fluctuations, is significantly difficult by contrast with short-range traffic flows [18]. Therefore, more features should be extracted from traffic flows in order to improve the effectiveness of anomaly detection in VANETs. In addition, a real-time anomaly detection algorithm is imperative corresponding to the characteristics of VANETs, which means that the anomaly detection algorithm with limited computational complexity is suitable for a VANET.

In this paper, we investigate the feasibility of applying Convolutional Neural Network (CNN) to extract some insightful features of network traffic, including spatial and temporal. A CNN architecture is designed to extract multi-dimensional features of all flows in a network at first. After that, network traffic is estimated via extracted features. Finally, a threshold-based method is adopted to identify anomalies from traffic dataset with hypothesized anomalies. The contributions of this paper can be summarized as follows:

- A CNN architecture is designed for traffic feature extraction. The traffic flows of a VANET not only yield temporal features but also spatial and spatio-temporal features. There are several explicit issues when we using CNN to extract these features, including the spatial dimension of convolution kernels and the number of convolutional layers. Hence, an appropriate CNN architecture is indispensable at first for feature extraction.
- A network traffic estimation approach is proposed in terms of the extracted features. As a threshold-based method, the accuracy of network traffic estimation is a crucial role which has an effect on the availability of anomaly detection. That is because the threshold is calculated via the estimation error. Thereby, the proposed CNN architecture should estimate network traffic precisely.

The rest organization of this paper is as follows. The related work is shown in Section II. The background of CNN and the details of the proposed method are described

in section III. In section IV, numerical experiments using real traffic are provided to evaluate the proposed method. We conclude our work in section V, and summarize the future work that can improve the scalability and effectiveness of our method.

II. RELATED WORK

In traditional ISP network, anomaly detection issues have been widely explored, and exhibited sufficient detection accuracy in a variety of scenarios [19]–[26]. These methods usually extracted the statistical features of network traffic (e.g., long-range dependence, short-range dependence and multifractal feature) at first, and then used a threshold-based method to identify anomalies. For instance, in [22], AsSadhan *et al.* took advantage of the long-range dependence extracted by the second-order self-similar (SOSS) and fractional autoregressive integrated moving average (FARIMA) models for anomaly detection. They first used the SOSS and FARIMA models to estimate an traffic flow. After that, the anomalies were identified by comparing the errors with a threshold. The network traffic obeys not only temporal features but also some interesting spatial features. Therefore, some methods based on spatial features of network traffic have been proposed to improve the false alarm rate such as the work in [23] and [24]. Moreover, machine learning techniques have been employed widely in ISP network anomaly detection discipline. The Principal Component Analysis (PCA) method was adopted in anomaly detection initially, which had to make use of a great deal of prior knowledge about link observations. In this case, it was significantly difficult to guarantee real-time detection. For this issue, in [19], a data density-based method was proposed, in which the data density was recursively calculated to identify anomalies via the distribution of metrics instead of a threshold as other methods. The proposed approach needs not any prior knowledge about anomalous behaviors. Besides, in [26], Simmross-Wattenberg *et al.* used α-stable model to fit network traffic, and identify anomalous behaviors by a threshold-based method. Despite lots of state-of-the-art anomaly detection algorithms have been proposed, however, these algorithms are not capable of employing in VANETs directly. Motivated by this issue, novel anomaly detection algorithms were explored for VANETs. In [27], Pang *et al.* extracted the moving station of vehicles and spatial features of jammers to detect jamming attacks. The deep learning has been proved that it is a powerful tool to extract the valuable features of network traffic, whereas the computational complexity is tremendous that satisfy short-lived link and dynamic topology in VANETs.

III. ANOMALY DETECTION METHOD

In this section, the details of our anomaly detection approach are introduced. The process of the proposed anomaly detection approach consists of two phases: 1) estimating traffic based on CNN; 2) detecting anomalies based on a threshold.

A. BACKGROUND OF CONVOLUTIONAL NEURAL NETWORK

As a well-known paradigm in deep neural networks, the CNN is ubiquitous in image processing and pattern recognition [28]–[32]. It derives from the redundant spatio-temporal features of images, and the powerful ability of a CNN in capturing the spatio-temporal features for the input testing datasets. The CNN captures a spatio-temporal feature of testing datasets by a convolution kernel. Moreover, a series of kernels can obtain various feature maps. Additionally, the sub-sampling layers are used to reduce computation time. For a convolutional layer in a neural network, each neuron is associated with a spatial location (i, j) with respect to the input maps. Meanwhile, each output map may combine convolutions with multiple input maps, that is

$$\boldsymbol{a}_{i,j}^{(l)} = f\left(\left(\boldsymbol{a}^{(l-1)}\boldsymbol{k}^{(l)}\right)_{i,j} + b^{(l)}\right). \tag{1}$$

The common activation functions $f(\cdot)$ are sigmoids and rectified linear units (ReLUs).

Besides, the classical backpropagation algorithm can learn a CNN efficiently. Due to the above advantages of CNN, we propose an anomaly detection approach based on CNN.

B. END-TO-END NETWORK TRAFFIC

The end-to-end network traffic exhibits the volume of traffic flows between possible origin and destination (OD) node pairs. The nodes can be defined as various granularities, including routers and point of presences and so on. This traffic information can be denoted by a matrix so-called traffic matrix (denoted by $\boldsymbol{X}$) which is a crucial input parameter in network management and traffic engineering. The element $x_{o,d,t}$ in a traffic matrix denotes the average of traffic entering a network at node o and leaving at node d over $[t, t + \Delta t]$. In this paper, an OD flow is denoted by $\boldsymbol{x}_n$ whose elements are $x_{n,t}$ and $n = 1, 2, \ldots, N$ for the network with $N^{\frac{1}{2}}$ nodes.

C. TRAFFIC MATRIX ESTIMATION

In our anomaly detection approach, we first estimate the traffic matrix according to the spatio-temporal features of traffic flows. The architecture of CNN for traffic matrix estimation is shown in Fig. 4. The consideration of this CNN architecture is the long-range and short-range dependence. Meanwhile, the timescale (that is the size of Δt) is also considered. This CNN architecture includes 8 hidden layers, that is, 4 convolutional layers and 4 sub-sampling layers with a factor of 2. The first three convolutional layers learn 6 convolution kernels of 7×7, 6×6, and 5×5 spatial dimensions, respectively. The last convolutional layer are built by 12 convolution kernels of 5×5 spatial dimensions. A fully connected layer transforms the high-level features of traffic matrix into the output layer with N elements for estimation.

In our approach, the input traffic matrix is square, and each element is denoted by $x_{n,t}$ where $n, t = 1, 2, \ldots, N$ (shown by Fig. 5). For the convolutional layer, the j-th element of the m-th kernel in the l-th convolutional layer is $k_j^{(l,m)}$, and

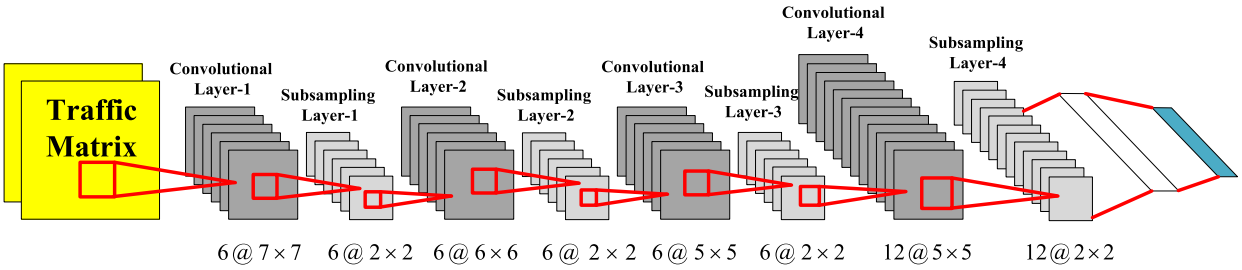


FIGURE 4. CNN architecture.

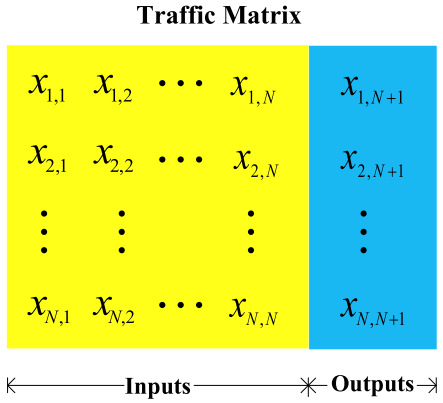


FIGURE 5. Training datasets.

$j = 1, 2, \ldots, N^{(l)}$, $m = 1, 2, \ldots, M^{(l)}$ where $N^{(l)}$ and $M^{(l)}$ are the number of elements and kernels in the l-th layer respectively. An individual neuron takes a vector of inputs $\boldsymbol{a}_q^{(l-1,m)}$ in the l-th convolutional layer (shown by Fig. 6), and

$$q = 1, 2, \ldots, \left(\frac{1}{2^l}N - \sum_{i=1}^{l}\frac{1}{2^i}\sqrt{N^{(i)}} + 1 - \frac{1}{2^l}\right)^2. \quad (2)$$

The output map is

$$a_q^{(l,m)} = sigmoid\left(\sum_{m'=1}^{M^{(l-1)}} \boldsymbol{a}_q^{(l-1,m')}\boldsymbol{k}^{(l,m)} + b^{(l,m)}\right), \quad (3)$$

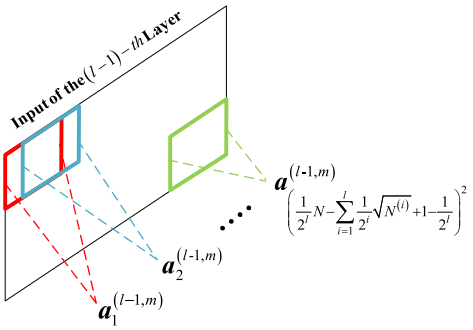


FIGURE 6. Input maps of the l-th convolutional layer.

where $b^{(l,m)}$ is the bias of the m-th kernel in the l-th convolutional layer. $sigmoid(\cdot)$ performs an operation known as the sigmoid function. The vector $\boldsymbol{k}^{(l,m)} = \left(k_1^{(l,m)}, k_2^{(l,m)}, \ldots, k_{N^{(l)}}^{(l,m)}\right)$ is the vectorial representation of kernels. All the sub-sampling layers perform the average pooling operation with respect to the factor 2. The output layer is an $N \times 1$ vector matching an estimator of traffic matrix over a time slot.

During the training process, the input and output training datasets are made up of consecutive network traffic shown in Fig. 5. To implement traffic estimation, the inputs and outputs are $N \times N$ matrix and $N \times 1$ vector, respectively. End-to-end network traffic has various statistical features including spatial and temporal. The spatial features of end-to-end network traffic derive from the correlation of network services and the location distribution of servers and users. Meanwhile, the traffic matrix is built by all possible origin-destination node pairs, hence the order of OD flows also has an effect on the learning of spatio-temporal features. Motivated by that, the OD flows in traffic matrix are ordered with respect to the correlation. Besides, network traffic is nonnegative and tremendous. Otherwise, the sigmoid function means that the sensitivity of an individual neuron depends on the independent variable over a tiny symmetric interval. As a result, most of neurons are usually dormant for network traffic. Therefore, the traffic matrix is preprocessed by centering and normalization in our approach. Centering and normalizing matrix means that each element of traffic matrix is $[-1, +1]$. Based on above preprocessing operation, given M training examples $\left(\left(\boldsymbol{X}^{(1)}, \boldsymbol{x}_{N+1}^{(1)}\right), \left(\boldsymbol{X}^{(2)}, \boldsymbol{x}_{N+1}^{(2)}\right), \ldots, \left(\boldsymbol{X}^{(M)}, \boldsymbol{x}_{N+1}^{(M)}\right)\right)$,the single loss function L_m is defined as a one-half squared-error, that is,

$$L_m = \frac{1}{2}\left\|h_{k,b}\left(\boldsymbol{X}^{(m)}\right) - \boldsymbol{x}_{N+1}^{(m)}\right\|^2, \quad (4)$$

where $h_{k,b}\left(\boldsymbol{X}^{(m)}\right)$ denotes an output of CNN with respect to the corresponding traffic matrix. The overall loss function with respect to m training examples is

$$L = \frac{1}{m}\sum_{m=1}^{M} L_m, \quad (5)$$

Algorithm 1 Calculate $y = x^n$

Require: M training examples; testing dataset $\boldsymbol{X}$; traffic dataset with anomalies $\boldsymbol{x}_{N+1}$

Ensure: Anomalies

$m \leftarrow 1$
$n \leftarrow 1$
$t \leftarrow 1$
while $m \leq M$ **do**
  while $n \leq N$ **do**
    while $t \leq N+1$ **do**
      $x_{n,t}^{(m)} \leftarrow x_{n,t}^{(m)} - mean\left(\boldsymbol{x}_n^{(m)}\right)$
      $x_{n,t}^{(m)} \leftarrow \frac{x_{n,t}^{(m)}}{\max\left(\boldsymbol{x}_n^{(m)}\right)}$
      $t \leftarrow t+1$
    end while
    $n \leftarrow n+1$
  end while
  $m \leftarrow m+1$
end while
Train convolutional neural network by backpropagation algorithm using training dataset;
Estimate network traffic $\hat{\boldsymbol{x}}_{N+1}$ by performing the feed-forward pass;
Perform anomaly detection by 3σ approach using $\hat{\boldsymbol{x}}_{N+1}$ and $\boldsymbol{x}_{N+1}$;

which is an average of the single loss function set $\{L_m\}_{m=1}^{M}$. The parameters consisting of kernels and biases are solved by gradient descent. Solving convolution kernels and biases is obtained by way of calculating the derivative

$$\begin{cases} \dfrac{\partial L}{\partial k_i^{(l,m)}} \\ \dfrac{\partial L}{\partial b^{(l,m)}} \end{cases} \tag{6}$$

aiming at minimizing the loss function L. Each parameter is updated with respect to the learning rate α as follows:

$$\begin{cases} k_i^{(l,m)} = k_i^{(l,m)} - \alpha \dfrac{\partial L}{\partial k_i^{(l,m)}} \\ b^{(l,m)} = b^{(l,m)} - \alpha \dfrac{\partial L}{\partial b^{(l,m)}} \end{cases} \tag{7}$$

Moreover, the derivatives can be computed by means of backpropagation.

D. ANOMALY IDENTIFICATION

The general approach of anomaly identification is to deal with the problem that which traffic out of normal behaviors. For this problem, a simple threshold-based identification approach is used to detect anomalies by way of separating network traffic estimation error, that is a 3σ deviation from the mean [32]–[34]. The details of our approach are shown as follows.

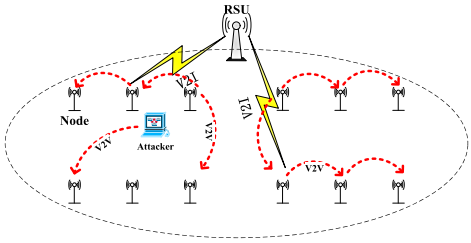


FIGURE 7. Architecture of testbed.

IV. NUMERICAL RESULTS

In this section, we assess the proposed anomaly detection algorithm (called CNN) by real network traffic from our testbed shown by Fig. 7. The testbed consists of a RSU and 12 nodes which can link each other by wireless links. Besides, to simulate the quick movement of nodes, we assume that the distance between two neighbor nodes obeys the Gaussian distribution whose expectation and variance are 10 and 5. Furthermore, the Open Shortest Path First (OSPF) method is adopted to generate the topology of the wireless network. In the OSPF method, the weights (denoted by $W_{i,j}$, where $i, j \in \{1, 2, \ldots, 12\}$) are defined by the general urban path loss model [35], that is

$$\begin{aligned} W_{i,j} &= P_{GUPL}\left(d_{i,j}\right) \\ &= -10\log_{10}\left(\left(\frac{\lambda}{4\pi d_0}\right)^{\beta}\right) + 10n\log_{10}\left(\frac{d_{i,j}}{d_0}\right) \\ &\quad + \alpha d_{i,j} + f_{FAF}, \end{aligned} \tag{8}$$

where $d_{i,j}$ denoted the distance between the nodes i and j, and d_0 ($d_0 \gg \lambda/2\pi$) is the close-in reference distance. β is the received power decays with distance at a rate of $10\beta dB/decade$. n and α are path loss exponent and attenuation constant, respectively. f_{FAF} is the floor attenuation factor. In our testbed, we use the Low Orbit Ion Cannon as an attacker, see Fig. 7. All network data packets are extracted by the Wireshark, and the network traffic is counted by the MATLAB. For comparison, the PCA method which is also a machine learning-based method is involved in our simulations. The PCA method uses the singular value decomposition and the squared prediction error to divide the link dataset into a normal subspace and an abnormal subspace at first. The abnormal subspace defines a set of potential anomalies. At last, the best hypothesis that can explain the largest amount of residual traffic is chosen from the potential anomalies.

A. TRAFFIC ESTIMATION

In this subsection, we first evaluate the performance of our method in network traffic estimation, and compare real network traffic with corresponding estimators directly. Figures 8 and 9 display two OD flows selected discretionarily from our testbed. The x-axis and y-axis denote the time slot and the volume of traffic flow, respectively. It is well known that estimating irregular fluctuations is a tough task for the network traffic estimation in traditional ISP networks.

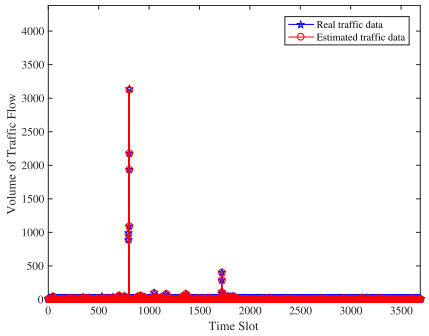


FIGURE 8. Real network traffic and their estimators.

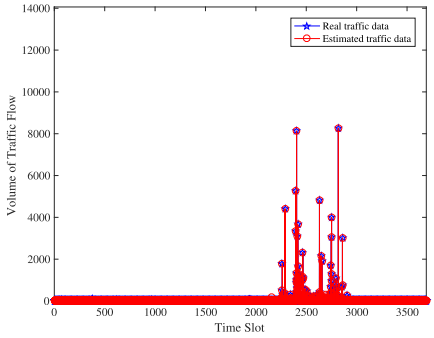


FIGURE 9. Real network traffic and their estimators.

From these figures, we find that the proposed CNN architecture is able to track the traces of OD flows exactly, thought these OD flows comprise many steep fluctuations.

After comparing the real network traffic with their estimators, we will validate the proposed method quantitatively. Thereby, the spatial relative errors (SREs) and temporal relative errors (TREs) are involved in our simulations as a metric, which is defined as

$$\begin{cases} SRE\ (n) = \dfrac{\left\|\hat{x}_{n,t} - x_{n,t}\right\|_2}{\left\|x_{n,t}\right\|_2} \\ TRE\ (t) = \dfrac{\left\|\hat{x}_{n,t} - x_{n,t}\right\|_2}{\left\|x_{n,t}\right\|_2}, \end{cases} \tag{9}$$

where $\hat{x}_{n,t}$ is the estimator of $x_{n,t}$, and $\|\cdot\|_2$ denotes the l_2-norm. Figures 10 and 11 display the cumulative distribution functions of SRE and TRE, respectively. As shown by Fig. 10, comparing with the PCA method, the SREs of the proposed method is significantly small. Moreover, Fig. 10 states that the SREs of about 90% of OD flows are less than 1.11. Likewise, it is 1.73 for the PCA method. Consider the TREs of two methods, from Fig. 11, we see that the CNN method obtains an explicit improvement for the PCA method. Obviously, for about 90% of moments, the TREs of CNN is less than 0.08. By contrast, it is 1.38 for the PCA method. The CNN method estimates the network traffic by extracting the spatio-temporal features of traffic matrix, thus both SRE and TRE are lower than that of PCA. In addition, the PCA method has significantly large TRE due to the irregular fluctuations of network traffic in VANETs.

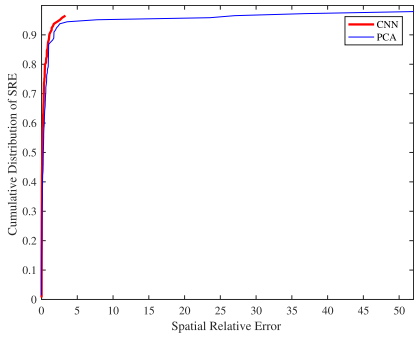


FIGURE 10. Cumulative distribution functions of SRE.

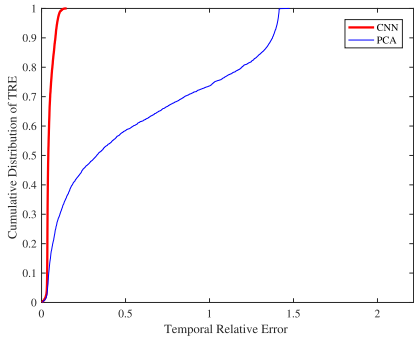


FIGURE 11. Cumulative distribution functions of TRE.

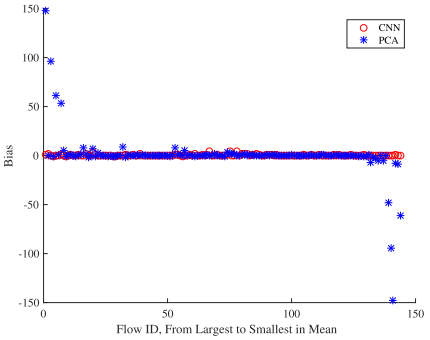


FIGURE 12. Estimation bias.

As well known, an unbiased estimator may have high variance, so that it cannot provide an accurate estimator that is close to the true value [36]. For this reason, we validate the estimation bias and their variance, which can be defined as

$$\begin{cases} B\ (n) = \dfrac{\sum_{t=1}^{T}\left(\hat{x}_{n,t} - x_{n,t}\right)}{T} \\ SD\ (n) = \sqrt{\dfrac{\sum_{t=1}^{T}\left(\left(\hat{x}_{n,t} - x_{n,t}\right) - B\ (n)\right)^2}{1-T}} \end{cases} \tag{10}$$

$SD\ (n)$ is so-called the sample standard deviation of the estimator, which can be viewed as a metric of the variance. In Fig. 12, we observe that it exhibits the biases of two methods. The x-axis shows the identities of OD flows arranged with respect to the means of them. We see that the PCA method has more or less biases for both lightweight and elephant flows. In Fig. 13, we find that CNN has lower

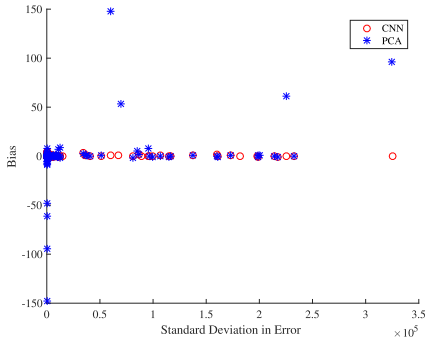


FIGURE 13. Estimation bias versus standard deviation.

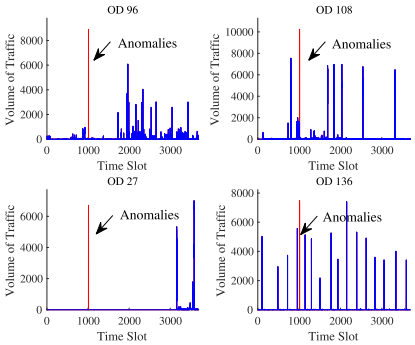


FIGURE 14. Traffic dataset with anomalies.

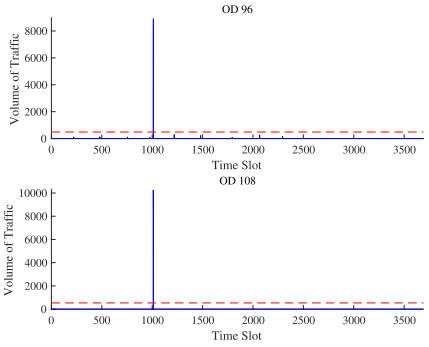


FIGURE 15. Anomaly detection for elephant flows.

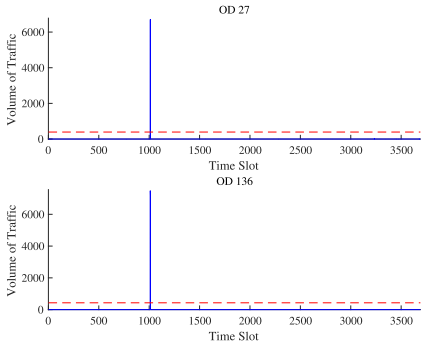


FIGURE 16. Anomaly detection for lightweight flows.

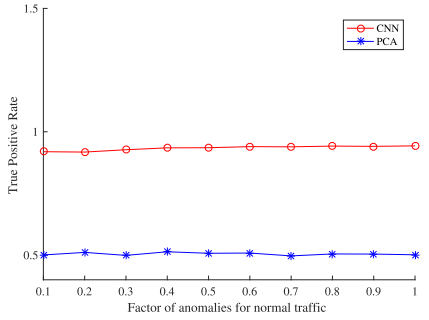


FIGURE 17. True positive rates of CNN and PCA with various factors.

variance than PCA, which means that CNN tends to exactly estimate short timescale variations.

B. ANOMALY DETECTION

Figure 14 shows the selected OD flows in our analysis from the dataset of the testbed. The x-axis denotes the time slot of network traffic, and the y-axis is the volume of corresponding traffic flows. The real traffic is mixed by anomalies as shown by four sub-figures. We first plot the anomaly detection results via the proposed approach. Four OD flows are selected according to the size of traffic flows. Figure 15 plots the anomaly detection results of elephant flows in the network. Relatively, Fig. 16 exhibits the performance of the proposed approach for the traffic with small volume. The red dotted line is the identity threshold by means of the 3σ deviation method for anomaly detection. Two figures indicate that our approach can detect the anomalies in traffic flows precisely.

Figure 17 displays true positive rate of two methods. In this simulation, a factor of anomalies with respect to normal traffic is defined, which is denoted by

$$\begin{cases} \boldsymbol{x}_{mix} = \boldsymbol{x}_0 + \boldsymbol{x}_{attacks} \\ f = \dfrac{\bar{\boldsymbol{x}}_{attacks}}{\bar{\boldsymbol{x}}_0}, \end{cases} \tag{11}$$

where $\boldsymbol{x}_0$ and $\boldsymbol{x}_{attacks}$ are normal and abnormal traffic. $\boldsymbol{x}_{mix}$ is the total of traffic consisting of normal and abnormal traffic. $\bar{\boldsymbol{x}}_{attacks}$ and $\bar{\boldsymbol{x}}_0$ are the means of normal and abnormal traffic. 3000 tests are also carried out under each factor, and the true positive rate is calculated via these 3000 tests.

Figure 17 shows the trace of true positive rates with respect to the factors. It indicates that CNN has high true positive rate comparing with the PCA method.

V. CONCLUSION AND FUTURE WORK

The spatio-temporal features of traffic matrix are taken into account in anomaly detection, and a CNN-based anomaly detection approach is proposed in this paper. The hierarchical convolutional and sub-sampling layers extract multifractal and low-rank features deeply for network traffic estimation. Additionally, a threshold-based method is referred into our approach for the further anomaly detection. By analyzing the proposed approach applying to real network dataset, the simulation results declare that the proposed method can identify the anomalies in normal traffic flows exactly for VANETs.

The further work is indispensable to improve the proposed approach in scalability and computational complexity. First, the CNN architecture is built (e.g., the number of kernels and convolutional layers) for a specific dataset in this paper. For popularization and application, an universal and self-adaptive CNN architecture is imperative, which is the main work in the future. Besides, on-line anomaly detection is a critical application. The CNN can decrease the cost of deep learning effectively, but it's still insufficient in on-line anomaly detection. Using a great number of historical training datasets also limits the performance for new network attacks. Hence, a lightweight learning algorithm is also a main work for improvement.

ACKNOWLEDGMENT

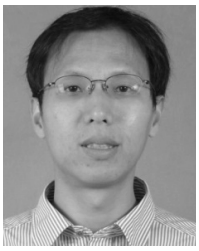
The authors wish to thank the reviewers for their helpful comments.

REFERENCES

[1] T. Huang, H. Sethu, and N. Kandasamy, "A new approach to dimensionality reduction for anomaly detection in data traffic," *IEEE Trans. Netw. Service Manage.*, vol. 13, no. 3, pp. 651–665, Sep. 2016.
[2] W. Hou, Z. Ning, L. Guo, Z. Chen, and M. S. Obaidat, "Novel framework of risk-aware virtual network embedding in optical data center networks," *IEEE Syst. J.*, to be published, doi: 10.1109/JSYST.2017.2673828.
[3] X. Ma and Y. Chen, "DDoS detection method based on chaos analysis of network traffic entropy," *IEEE Commun. Lett.*, vol. 18, no. 1, pp. 114–117, Jan. 2014.
[4] D. He, S. Chan, X. Ni, and M. Guizani, "Software-defined-networking-enabled traffic anomaly detection and mitigation," *IEEE Internet Things J.*, vol. 4, no. 6, pp. 1890–1898, Dec. 2017.
[5] J. Wang and I. C. Paschalidis, "Botnet detection based on anomaly and community detection," *IEEE Trans. Control Netw. Syst.*, vol. 4, no. 2, pp. 392–404, Jun. 2017.
[6] L. Xiao, Y. Li, G. Han, G. Liu, and W. Zhuang, "PHY-layer spoofing detection with reinforcement learning in wireless networks," *IEEE Trans. Veh. Technol.*, vol. 65, no. 12, pp. 10037–10047, Dec. 2016.
[7] Z. Ning, X. Wang, X. Kong, and W. Hou, "A social-aware group formation framework for information diffusion in narrowband Internet of Things," *IEEE Internet Things J.*, vol. 5, no. 3, pp. 1527–1538, Jun. 2018.
[8] J. Zhang and I. C. Paschalidis, "Statistical anomaly detection via composite hypothesis testing for Markov models," *IEEE Trans. Signal Process.*, vol. 66, no. 3, pp. 589–602, Feb. 2018.
[9] I. Nevat *et al.*, "Anomaly detection and attribution in networks with temporally correlated traffic," *IEEE/ACM Trans. Netw.*, vol. 26, no. 1, pp. 131–144, Feb. 2018.
[10] Y. Wang, Z. Wu, Q. Li, and Y. Zhu, "A model of telecommunication network performance anomaly detection based on service features clustering," *IEEE Access*, vol. 5, pp. 17589–17596, 2017.
[11] B. Wang, F. Hu, Y. Zhao, and T. N. Guo, "Anomaly detection and array diagnosis in wireless networks with multiple antennas: Framework, challenges and tools," *IEEE Netw.*, vol. 32, no. 1, pp. 152–159, Jan./Feb. 2018.
[12] X. Wang, Z. Ning, and L. Wang, "Offloading in Internet of vehicles: A fog-enabled real-time traffic management system," *IEEE Trans. Ind. Informat.*, to be published, doi: 10.1109/TII.2018.2816590.
[13] A. Boualouache, S.-M. Senouci, and S. Moussaoui, "A survey on pseudonym changing strategies for vehicular ad-hoc networks," *IEEE Commun. Surveys Tuts.*, vol. 20, no. 1, pp. 770–790, 1st Quart., 2018.
[14] P. K. Hoong, I. Tan, and C. Y. Keong, "BitTorrent network traffic forecasting with ARIMA," *Int. J. Comput. Networks Commun.*, vol. 4, no. 4, pp. 143–156, 2012.
[15] M. Akintunde, P. Kgosi, and D. Shangodoyin, "Evaluation of GARCH model adequacy in forecasting non-linear economic time series data," *J. Comput. Modell.*, vol. 3, no. 2, pp. 1–20, 2013.
[16] Z. Ning, J. Huang, and X. Wang, "Vehicular fog computing: Enabling real-time traffic management for smart cities," *IEEE Wireless Commun.*, to be published.
[17] M. Joshi and T. H. Hadi. (2015). "A review of network traffic analysis and prediction techniques." [Online]. Available: https://arxiv.org/abs/1507.05722
[18] Y. Liu, B. Li, X. Sun, and Z. Zhou, "A fusion model of SWT, QGA and BP neural network for wireless network traffic prediction," in *Proc. IEEE Int. Conf. Commun. Technol.*, Nov. 2013, pp. 769–774.
[19] S. N. Shirazi, S. Simpson, A. Gouglidis, A. Mauthe, and D. Hutchison, "Anomaly detection in the cloud using data density," in *Proc. IEEE 9th Int. Conf. Cloud Comput. (CLOUD)*, Jun./Jul. 2016, pp. 616–623.
[20] X. Hu *et al.*, "Emotion-aware cognitive system in multi-channel cognitive radio ad hoc networks," *IEEE Commun. Mag.*, vol. 56, no. 4, pp. 180–187, Apr. 2018.
[21] Z. Ning *et al.*, "A cooperative quality-aware service access system for social Internet of vehicles," *IEEE Internet Things J.*, to be published, doi: 10.1109/JIOT.2017.2764259.
[22] B. AsSadhan, K. Zeb, J. Muhtadi, and S. Alshebeili, "Anomaly detection based on LRD behavior analysis of decomposed control and data planes network traffic using SOSS and FARIMA models," *IEEE Access*, vol. 5, p. 13501–13519, 2017.
[23] P. Chhabra, C. Scott, E. D. Kolaczyk, and M. Crovella, "Distributed spatial anomaly detection," in *Proc. IEEE 27th Conf. Comput. Commun. (INFOCOM)*, Apr. 2008, pp. 2378–2386.
[24] I. C. Paschalidis and G. Smaragdakis, "Spatio-temporal network anomaly detection by assessing deviations of empirical measures," *IEEE/ACM Trans. Netw.*, vol. 17, no. 3, pp. 685–697, Jun. 2009.
[25] Z. Ning, X. Kong, F. Xia, W. Hou, and X. Wang, "Green and sustainable cloud of things: Enabling collaborative edge computing," *IEEE Commun. Mag.*, to be published.
[26] F. Simmross-Wattenberg, J. I. Asensio-Perez, P. Casaseca-de-la-Higuera, M. Martin-Fernandez, I. A. Dimitriadis, and C. Alberola-Lopez, "Anomaly detection in network traffic based on statistical inference and alpha-stable modeling," *IEEE Trans. Depend. Sec. Comput.*, vol. 8, no. 4, pp. 494–509, Jul./Aug. 2011.
[27] L. Pang, P. Guo, X. Chen, J. Li, and Z. Xue, "Estimating the number of multiple jamming attackers in vehicular ad hoc network," in *Proc. 6th Int. Conf. Comput. Sci. Netw. Technol. (ICCSNT)*, Oct. 2017, pp. 366–370.
[28] S. Yu, S. Jia, and C. Xu, "Convolutional neural networks for hyperspectral image classification," *Neurocomputing*, vol. 219, pp. 88–98, Jan. 2017.
[29] A. Jalali, R. Mallipeddi, and M. Lee, "Sensitive deep convolutional neural network for face recognition at large standoffs with small dataset," *Expert Syst. Appl.*, vol. 87, pp. 304–315, Nov. 2017.
[30] W. Huang, G. Song, H. Hong, and K. Xie, "Deep architecture for traffic flow prediction: Deep belief networks with multitask learning," *IEEE Trans. Intell. Transp. Syst.*, vol. 15, no. 5, pp. 2191–2201, Oct. 2014.
[31] W. Hou, Z. Ning, L. Guo, and X. Zhang, "Temporal, functional and spatial big data computing framework for large-scale smart grid," *IEEE Trans. Emerg. Topics Comput.*, to be published, doi: 10.1109/TETC.2017.2681113.
[32] C. L. P. Chen, C.-Y. Zhang, L. Chen, and M. Gan, "Fuzzy restricted Boltzmann machine for the enhancement of deep learning," *IEEE Trans. Fuzzy Syst.*, vol. 23, no. 6, pp. 2163–2173, Dec. 2015.
[33] D. Jiang, Z. Xu, P. Zhang, and T. Zhu, "A transform domain-based anomaly detection approach to network-wide traffic," *J. Netw. Comput. Appl.*, vol. 40, no. 2, pp. 292–306, Apr. 2014.
[34] Z. Ning, F. Xia, N. Ullah, X. Kong, and X. Hu, "Vehicular social networks: Enabling smart mobility," *IEEE Commun. Mag.*, vol. 55, no. 5, pp. 49–55, May 2017.
[35] J. Andrusenko, R. L. Miller, J. A. Abrahamson, N. M. M. Emanuelli, R. S. Pattay, and R. M. Shuford, "VHF general urban path loss model for short range ground-to-ground communications," *IEEE Trans. Antennas Propag.*, vol. 56, no. 10, pp. 3302–3310, Oct. 2008.
[36] A. Soule *et al.*, "Traffic matrices: Balancing measurements, inference and modeling," in *Proc. SIGMETRICS*, 2005, pp. 362–373.

LAISEN NIE is currently an Associate Professor with the School of Electronics and Information, Northwestern Polytechnical University, Xi'an, China. His research interests include network measurement, network security, and cognitive networks.

YONGKANG LI was born in Shaoyang, Hunan, China, in 1988. He received the B.Eng. degree in electronic information engineering and the Ph.D. degree in signal and information processing from Xidian University, Xian, China, in 2011 and 2016, respectively. From 2015 to 2016, he was a Visiting Ph.D. Student with the School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore. Since 2017, he has been an Assistant Professor with the School of Electronics and Information, Northwestern Polytechnical University, China. His research interest is radar signal processing and network security.



XIANGJIE KONG received the Ph.D. degree from Zhejiang University, Hangzhou, China. He is currently an Associate Professor with the School of Software, Dalian University of Technology, China. He has published over 50 scientific papers in international journals and conferences. His research interests include big traffic data, social computing, and mobile computing.